Un système de reconnaissance de mots arabes manuscrits hors-ligne sans signes diacritiques

F. Menasri, N. Vincent, E. Augustin, M. Cheriet

Un système de reconnaissance de mots arabes manuscrits hors-ligne sans signes diacritiques

Farès Menasri[1,2] – Nicole Vincent[1] – Emmanuel Augustin[2] – Mohamed Cheriet[3]

[1] Université PARIS DESCARTES - UFR de Mathématiques et Informatique
45 rue des Saints Pères, 75006 Paris - FRANCE

[2] A2iA SA
40 bis rue Fabert, 75007 Paris - FRANCE

[3] ETS Montréal - Laboratoire LIVIA
1100 rue Notre-Dame ouest , Montréal, QC, H3C 1K3 - CANADA

{fm,ea}@a2ia.com , nicole.vincent@math-info.univ-paris5.fr, mohamed.cheriet@etsmtl.ca

Résumé : *Nous désignons par "signes diacritiques" les points et autres composantes secondaires associées aux lettres. En arabe, les signes diacritiques sont indispensables pour différentier certaines lettres ou groupes de lettres. Néanmoins, nous montrons que cette information n'est pas systématiquement nécessaire dans une application de reconnaissance de l'écriture (nous travaillons ici sur la base IFN/ENIT [PEC 02], sur une tâche de classification de noms de villes tunisiennes).*

Dans cet article, nous présentons un système de reconnaissance hors-ligne de l'écriture arabe manuscrite. La segmentation en graphèmes et l'extraction de primitives sont les mêmes que celles précédemment développées pour la reconnaissance de l'écriture manuscrite latine. Nous reviendrons rapidement sur l'alphabet de corps de lettres présenté dans [MEN 07], et insisterons davantage sur les prétraitements spécifiques à l'arabe que nous avons mis en oeuvre pour adapter une chaîne de reconnaissance de l'écriture latine vers l'écriture arabe.

Mots-clés : Ecriture arabe manuscrite, Modèles de Markov cachés, Graphèmes.

1 Introduction

Bien que l'arabe soit une langue parlée par plus de 250 millions de personnes dans le monde, il n'y a pas à ce jour de système industriel de reconnaissance automatique de l'écriture arabe manuscrite. Les champs d'applications sont pourtant nombreux : automatisation du tri du courrier postal, du traitement des chèques, du traitement de formulaires, indexation automatique de manuscrits anciens, etc . . .

De nombreux travaux ont été menés au cours des dernières années [LOR 06, CHE 07], mais ce sujet reste un domaine de recherche actif.

L'apparition de bases publiques de tailles conséquentes comme la base IFN/ENIT [PEC 02] (24659 images, pour un vocabulaire de 937 de noms de villes tunisiennes, et 411 scripteurs), et l'organisation de compétitions en 2005 [MÄR 05] et en 2007 [MÄR 07] dans le cadre de la conférence ICDAR, ont rendu possibles les comparaisons entre systèmes et ont permis une progression rapide du domaine au cours des dernières années. Aujourd'hui, les performances obtenues par les meilleurs systèmes de reconnaissance de l'écriture manuscrite arabe semblent proches de celles qu'on pourrait attendre sur du latin pour une taille de vocabulaire équivalente.

Il est intéressant de noter que certains des systèmes qui rapportent de très bonnes performances s'appuient sur des systèmes précédemment développés pour la reconnaissance de l'écriture latine cursive (SIEMENS ou MIE dans [MÄR 07], ou encore [MÄR 06]), et qui sont parvenus à adapter leur chaîne de traitement pour la langue arabe. Ce résultat suggère que la reconnaissance de l'écriture arabe n'est pas un problème fondamentalement plus complexe que la reconnaissance de l'écriture cursive latine : moyennant quelques adaptations spécifiques à l'écriture arabe, il est possible de réutiliser certains modules précédemment développés pour la reconnaissance de l'écriture latine.

C'est également le parti que nous prenons : le système de reconnaissance de l'écriture arabe que nous avons mis au point s'appuie sur [KNE 98]. Il s'agit d'un système hybride à base de Modèles de Markov cachés et de Perceptron Multi-Couches, qui utilise une segmentation en graphèmes. Le schéma général du sytème de reconnaissance est donné figure 1. La partie reconnaissance des signes diacritiques n'est pas décrite dans cet article. Nous nous concentrerons sur la partie reconnaissance sans diacritiques.

Après un bref rappel de l'intérêt de l'alphabet de corps de lettres proposé dans [MEN 07] (section 2), nous présenterons de façon plus détaillée la procédure de détection des signes diacritiques (section 3), puis la détection de la bande de base (section 4). La segmentation en graphèmes et l'extraction de primitives (section 5) sont les mêmes que celles utilisées pour la reconnaissance de l'écriture manuscrite latine. Le reconnaisseur est également du même type que celui utilisé pour la reconnaissance de l'écriture latine. Son initialisation est discutée dans la section 6. Dans la section 7, nous donnerons les résultats expérimentaux obtenus sur la base IFN/ENIT. Et enfin, nous terminerons par les conclusions et perspectives (section 8).

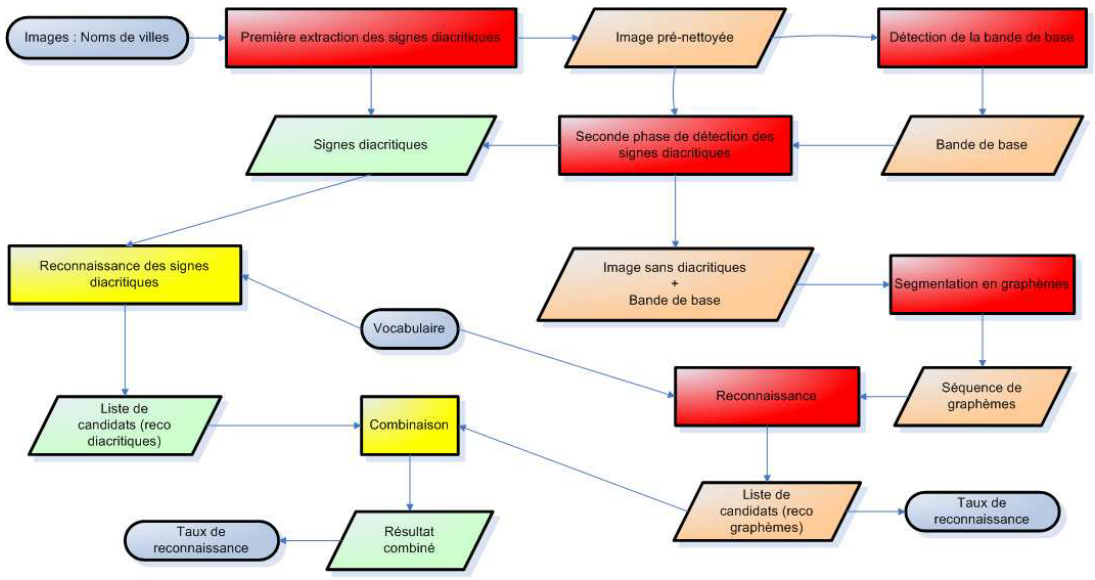


FIG. 1 – Système de reconnaissance de l'écriture arabe.
En rouge : étapes du système de reconnaissance sans diacritiques. En orange : données intermédiaires du système de reconnaissance sans diacritiques. En jaune : étapes supplémentaires pour la reconnaissance des diacritiques. En vert : données intermédiaires utilisées pour la reconnaissance avec diacritiques. En bleu : entrées et sorties du système.
Remarque : La partie reconnaissance des diacritiques et la partie combinaison sont données à titre indicatif, mais ne sont pas traitées dans cet article

2 Alphabet de corps de lettres

Un nouvel alphabet de corps de lettres a été présenté dans [MEN 07]. Cet alphabet repose sur trois constatations :

- la plupart des lettres arabes s'écrivent sous la forme radical + terminaison (voir paragraphe 2.1)
- certaines lettres ne se différentient que par le nombre et/ou la position des points (voir paragraphe 2.2)
- les ligatures verticales sont complexes à segmenter (voir paragraphe 2.3)

2.1 Un radical et une terminaison

On a souvent coutume de dire que les lettres arabes peuvent prendre 4 formes différentes en fonction de leur position dans le mot (début, fin, milieu et isolée). C'est vrai pour les lettres (ع ـع ـعـ عـ), (غ ـغ ـغـ غـ) et (ه ـه ـهـ هـ). Mais c'est faux dans le cas général.

Pour la plupart des lettres de l'alphabet arabe, les formes début et milieu sont identiques (à la ligature avec la lettre précédente près). Il en va de même pour les formes fin et isolée. De plus, les formes fin/isolée sont souvent construites à partir des formes début/milieu auxquelles on rajoute une "jambe" (voir tableau 1).

Jambe 1 : ـب	Jambe 2 : ـن	Jambe 3 : ـح
ب ت ث → ب	ل → لـ	ح → حـ
و → ف	س → سـ	ع → عـ
	ص → صـ	ع → عـ
	ن → نـ	

TAB. 1 – Lettres arabe : début/milieu vers isolée/fin

2.2 Des formes identiques aux points près

Certains groupes de lettres ne se différentient que par le nombre et/ou la position de leurs signes diacritiques (voir tableau 2).

د → {د ذ}	ر → {ر ز ژ ڕ}	س → {س ش}
ص → {ص ض}	ر → {ر ز}	ح → {ج ح خ}
و → {ف ق}	ط → {ط ظ}	ع → {ع غ}
ع → {ع غ}	ى → {ى ي}	ه → {ه ة}

TAB. 2 – Quelques lettres arabes et leur forme sans diacritiques

2.3 Les ligatures verticales

Les "ligatures verticales" sont des superpositions verticales de lettres (voir figure 2). Ces symboles peuvent être reconnus tels quels.

لا لحـ لحـ لـ

FIG. 2 – Ligatures verticales (sans les points) utilisées dans notre alphabet

2.4 Alphabet utilisé

Au final, nous utilisons l'alphabet de symboles décrit dans le tableau 3.

Dans [MEN 07], nous avons montré que l'utilisation de cet alphabet permettait d'améliorer les performances de reconnaissance. D'une part, en regroupant les classes de lettres similaires, les modèles qui les représentent disposent de plus

ا	ر	ه	ه	حـ	ح	د	ر	سـ	س
صـ	ص	ط	ء	عـ	ع	ع	و	كـ	ل
ل	مـ	م	ں	هـ	ھ	و	ى	Jambe 1 : ب	
لا	لـ	لم	لھ	ل					

TAB. 3 – Liste des symboles de l'alphabet proposé.

d'exemples en apprentissage. D'autre part, le fait de considérer les ligatures verticales en tant que classes à part entière, on évite la segmentation hasardeuse de ces groupes de symboles.

2.5 Confusions et ambiguïtes

Dans [MEN 07], nous avons montré que le fait d'ignorer les signes diacritiques et d'utiliser notre alphabet de corps de lettres, n'introduisait une ambiguité que sur 0,3% de la base IFN/ENIT, ce qui est négligeable par rapport aux taux d'erreurs rapportés sur cette base.

3 Détection des signes diacritiques

La détection des signes diacritiques a deux utilités. D'une part, un trop grand nombre de signes diacritiques risque de perturber l'histogramme de projection horizontale, qui sert a évaluer la bande de base (l'évaluation de la bande de base est détaillée dans la section 4). Et d'autre part, les signes diacritiques ainsi détectés pourront par la suite être utilisés pour améliorer les résultats de la reconnaissance. Ce deuxième point n'est pas abordé dans cet article.

La détection des signes diacritiques se fait en deux étapes. La première étape consiste à effectuer un filtrage des composantes connexes en s'appuyant sur des critères assez simples : taille des boîtes englobantes, aire, superposition verticale. L'objectif est de rejeter la plupart des signes diacritiques, sans rejeter de composante connexe correspondant à un corps de lettre ou à un pseudo-mot. La figure 3 décrit cette procédure de filtrage. Les seuils sont fixés empiriquement.

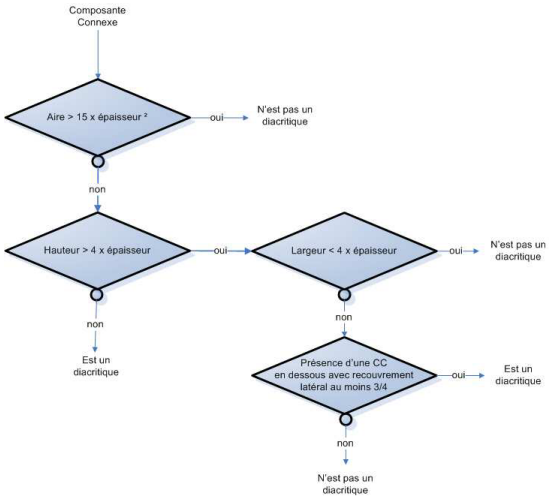


FIG. 3 – Procédure de filtrage des signes diacritiques : des seuils heuristiques définis à partir de l'épaisseur du tracé permettent d'effectuer un premier filtrage grossier.

Le premier test se base sur l'aire : les composantes connexes trop grosses ne peuvent pas être des signes diacritiques.

Les deuxième et troisième tests permettent de conserver les barres de alif (ا), qui contiennent peu de pixels et sont étendues verticalement.

Le 4ème test permet de conserver les caractères د isolés, qui sont parfois facilement assimilables à des signes diacritiques par leur forme. Cette règle repose sur la position relative de la composante connexe et de ses voisines : si une composante connexe C_1 d'aire réduite est située au-dessus d'une autre composante connexe C_2, et que C_2 recouvre verticalement C_1 à plus de 75%, alors C_1 est un signe diacritique.

Ce premier filtrage permet de détecter et d'écarter un certain nombre de composantes connexes. On peut alors réaliser une approximation de la bande de base sur l'image ainsi filtrée (voir section 4). Cette estimation de la bande de base permet d'effectuer un second filtrage plus précis.

Ce second filtrage prend en compte le fait que les signes diacritiques se situent soit en dessous, soit au-dessus de la bande de base.

Par exemple, dans la figure 4, une chadda n'a pas été filtrée lors de la première étape. Elle n'est pas supprimée car son recouvrement vertical n'est que partiel (inférieur au seuil fixé empiriquement) avec la lettre ر à laquelle elle est associée. Cette composante connexe a donc été conservée lors de la première étape. Mais après l'évaluation de la bande de base, il apparaît clairement qu'il s'agit bien d'un signe diacritique, puisque cette composante connexe est entièrement au-dessus de la bande de base. On peut donc l'ajouter à la liste des signes diacritiques détectés lors de la première étape.

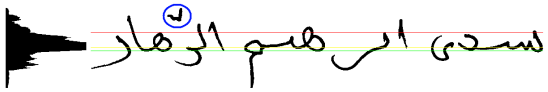


FIG. 4 – Pic maximal de l'histogramme (ligne jaune) et seuils pour déterminer une approximation de la ligne haute et de la ligne basse. On se sert de cette approximation de la bande de base pour retirer les signes diacritiques résiduels (ici une chadda entourée en bleu).

Dans certains cas, il arrive que les signes diacritiques ne forment pas des composantes connexes séparées du corps du texte. Un exemple est donné figure 5. Ces cas nécessitent la mise en oeuvre de mécanismes plus complexes, qui permettraient de générer plusieurs alternatives. Ce type de problème, relativement marginal, n'est pas traité ici.

4 Extraction de la bande de base

On utilise un histogramme de projection horizontale pour déterminer une première approximation de la bande de base. Pour éviter que cet histogramme ne soit perturbé par la présence d'un trop grand nombre de signes diacritiques, on effectue au préalable un filtrage de ces composantes (voir section 3).

Pour éviter les pics parasites qui peuvent apparaître sous la bande de base, nous utilisons les boucles pour restreindre

(a) Image initiale (b) Image obtenue après application de l'algorithme de suppression des signes diacritiques.

FIG. 5 – Présence d'un signe diacritique collé au corps de texte (chadda). Ces cas de figure ne sont pas traités.

la zone de recherche du maximum de l'histogramme. Cette partie sera présentée dans le paragraphe 4.1.

Une fois la première approximation de la bande de base obtenue à l'aide de seuils sur l'histogramme (voir paragraphe 4.2), on utilise le squelette pour déterminer plus précisément la ligne support de l'écriture (voir paragraphe 4.3).

4.1 Les boucles

En arabe, les descendants peuvent avoir des portions horizontales suffisamment longues pour perturber l'histogramme, et faire apparaître un pic parasite. La figure 6 illustre ce phénomène : en raison d'un grand nombre de descendants, le maximum de l'histogramme (pic principal) se situe sous la bande de base.

Une fois la plupart des signes diacritiques supprimés, on pré-localise la bande de base en utilisant les boucles, s'il y en a. En arabe, les boucles se situent dans la bande de base (ة , و ,ه ,ه ,ه ,ق ,ف ,ع ,ظ ,ط ,ض ,ص), à l'exception de la lettre (ﻫ) pour laquelle les boucles dépassent légèrement en haut et en bas. Localiser les boucles permet donc de déterminer une zone candidate pour la bande de base.

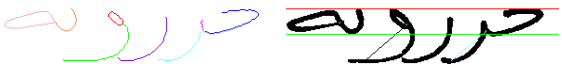
Mais en s'appuyant sur la localisation des boucles dans l'image, on définit une zone élargie centrée sur ces boucles (qui sont supposées dans la bande de base). C'est dans cette zone élargie qu'on recherche le maximum.

FIG. 6 – Utilisation des boucles pour déterminer une zone de recherche du maximum de l'histogramme de projection horizontale. Cette zone de recherche est une bande horizontale de trois fois la hauteur maximale des boucles, centrée sur les boucles. La recherche du maximum dans cette zone permet d'éviter le pic du bas, qui est dû à une succession de descendants.

4.2 Histogramme horizontal

A partir du maximum de l'histogramme, on applique un seuil haut et un seuil bas, qui permettent de déterminer une approximation de la bande de base (lignes rouge et verte de la figure 4).

Empiriquement, ces seuils sont fixés à :
seuil haut = 20% de la valeur du pic principal
seul bas = 45% de la valeur du pic principal.



(a) Extraction du squelette et des traits, pour déterminer les points singuliers (b) Détection des descendants

FIG. 7 – Utilisation des points singuliers du squelette pour trouver une évaluation plus fine de la bande de base.

Cette approximation de la bande de base permet d'effectuer un deuxième filtrage des composantes connexes pour achever la détection des signes diacritiques résiduels (deuxième étape de filtrage des signes diacritiques, décrite dans la section 3, un exemple est donné figure 4).

La hauteur de l'écriture (qui sera la hauteur de la bande de base précise) est supposée constante sur l'ensemble de la ligne. Elle est déterminée par la moyenne des distances verticales entre les extremums locaux du contour situés dans cette première estimation de la bande de base (entre les lignes rouge et verte de la figure 4).

4.3 Squelette

Le squelette est extrait à l'aide d'une version modifiée de l'algorithme de Hilditch [HIL 69], utilisée pour la reconnaissance de l'écriture latine. Une image du squelette est donnée figure 7(a).

Les points singuliers du squelette sont les points dont le voisinage est supérieur à 2. Ce sont les points d'embranchement, points de croisement ou boucles. Ils sont marqués en vert dans la figure 7(b). Certains de ces points singuliers servent d'estimateurs locaux plus précis de la position de la ligne d'écriture (ligne de base).

Les minima globaux des arcs de squelette sont également extraits. Les points singuliers retenus sont ceux qui se situent dans la bande de base, et dont l'un des arcs de squelette auxquels ils sont reliés ont un minimum global qui se situe sous la bande de base.

Un exemple est donné figure 7(b). Dans cet exemple, trois points singuliers sont candidats.

- Celui de gauche appartient à une boucle. Il n'est pas retenu, car les arcs de squelette auxquels il est relié ne vont pas sous la bande de base.
- Celui du milieu appartient également à une boucle. Il est retenu, un trait noir est marqué pour le relier au minimum global de l'arc de squelette considéré.
- Celui de droite appartient à un embranchement. Il est retenu, un trait bleu est marqué pour le relier au minimum global de l'arc de squelette considéré.

Les minima locaux du contour inférieur sont également retenus s'ils se situent dans la bande de base. Ces minima locaux et les points singuliers retenus sont de bons estimateurs locaux de la position de la ligne de base. La ligne de base affinée est donc construite comme une interpolation linéaire à partir de ces points retenus. La partie supérieure de la bande de base est donnée par la hauteur du tracé, supposée constante sur toute la ligne.

Au final, on obtient une détection de la ligne de base qui s'adapte bien aux légères variations de l'inclinaison de l'écri-

ture au sein d'une même ligne (voir figure 8).

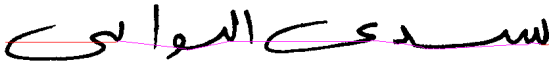
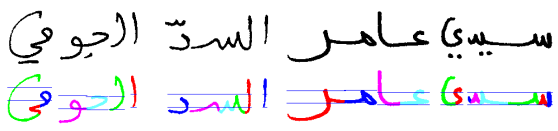


FIG. 8 – Détection fine de la bande de base.

5 Segmentation et extraction de primitives

La segmentation en graphèmes utilisée dans ce système est la même que celle utilisée dans la reconnaissance de l'écriture cursive latine [DUP 03].

Des exemples de segmentation en graphèmes sont donnés figure 9.



(a) السدّ الجوفي (b) سيدي عا مر

FIG. 9 – Exemples de segmentations en graphèmes.

La séquence de graphèmes est parcourue de droite à gauche, dans le sens de la lecture. Sur chaque graphème, un vecteur de caractéristiques de 74 dimensions est extrait. Cette extraction de primitives est la même que celle utilisée pour l'écriture latine.

6 Reconnaisseur et initialisation

Le reconnaisseur utilisé est du même type que celui utilisé pour la reconnaissance de l'écriture latine. Il s'agit d'un système hybride à base de Modèles de Markov Cachés (MMC) et d'un Réseau de Neurones (RN) de type Perceptron MultiCouches [KNE 98]. Les états sont regroupés en colonnes, et chaque état ne peut émettre qu'une seule classe d'observations (voir figure 10). Les probabilités d'émissions des MMC sont déléguées au RN.

Les modèles de mots sont construits par la concaténation des modèles de lettres qui les composent.

L'apprentissage des MMC et du RN se fait de manière séparée. Il s'agit d'un apprentissage de type Baum-Welch en mode batch en quatre étapes :

1. Décoder les bases de mots avec le système RN + MMC pour créer une base de vecteurs caractéristiques annotés pour le RN.
2. Entraîner le RN par rétropropagation du gradient (gradient stochastique).
3. Utiliser le nouveau RN pour calculer les probabilités d'observation.
4. Optimiser les probabilités de transition des états des MMC par l'algorithme de Baum-Welch.

Ce processus itératif est répété sur l'ensemble de la base d'apprentissage jusqu'à saturation des performances.

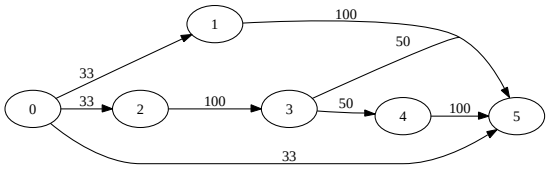


FIG. 11 – Topologie d'un modèle de lettres. Les transitions sont initialisées de manière uniforme.

Pour initialiser le RN, on commence par effectuer une partition non supervisée (k-Means) sur l'ensemble des vecteurs de primitives extraits à partir des graphèmes. On utilise ce k-Means pour annoter le RN, de telle sorte que ce dernier reproduise la fonction de classification du k-Means. Les MMC et le RN ainsi initialisés sont ensuite entraînés tour à tour selon la procédure itérative décrite ci-dessus.

7 Résultats expérimentaux

Le système est évalué sur la base IFN/ENIT, selon le protocole expérimental proposé dans [PEC 02] : apprentissage sur {a b c} et test sur { d }.

Le tableau 4 donne les performances de notre système et compare ces performances avec les autres systèmes ayant publiés des résultats en suivant le même protocole expérimental.

Les résultats montrent que la stratégie décrite dans cet article pour déterminer la bande de base donne de meilleures performances que l'approche naïve qui consisterait à choisir une bande de base horizontale déterminée par le pic de l'histogramme. Ces résultats montrent également que les signes diacritiques ne sont pas indispensables pour obtenir des performances proches de l'état de l'art. En revanche, le fait d'écarter volontairement les signes diacritiques prive le système d'une partie de l'information : bien que les signes diacritiques ne soient pas indispensables, ils apportent une information pertinente. Une combinaison adéquate permet d'améliorer les performances. Cette combinaison n'est pas présentée ici, et fera l'objet d'une publication ultérieure.

8 Conclusions et perspectives

Dans la section 2, nous avons brièvement rappelé l'idée d'utiliser un alphabet de corps de lettres [MEN 07] pour améliorer la reconnaissance.

En adaptant les prétraitements (sections 3 et 4), nous avons montré qu'il était possible de s'appuyer sur un système de reconnaissance précédemment développé pour l'écriture cursive latine, à base de système hybride MMC + RN avec une segmentation en graphèmes [KNE 98].

Bien que les signes diacritiques ne soient pas indispensables pour effectuer une tâche de reconnaissance de noms de villes sur la base IFN/ENIT, ces symboles contiennent néanmoins une information pertinente. Nous travaillons actuellement sur la combinaison entre le reconnaisseur sans diacritiques présenté dans cet article, et un reconnaisseur de diacritiques. Plusieurs stratégies sont possibles. Les premiers résultats sont encourageants : une première version, qui fera l'objet d'une publication ultérieure, permet d'améliorer significativement les performances en réduisant le taux d'er-

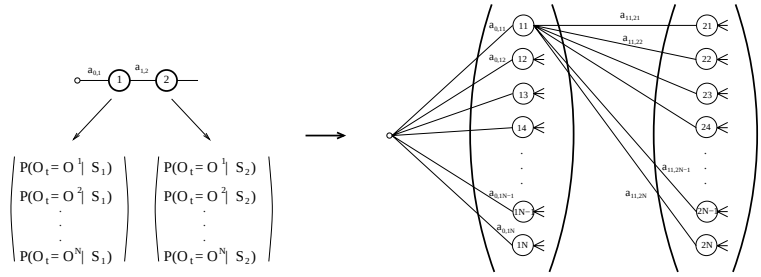


FIG. 10 – MMC à N classes d'observations : modèle standard et modèle hybride.

	1ère pos	2ème pos	10ème pos
Apprentissage {a,b,c} sans diacritiques	92.92	95.60	98.15
Test {d} : Bande de base simple (max histogramme)	84.72	89.84	95.74
Test {d} : notre système sans diacritiques	**89.98**	**93.54**	**97.45**
Test {d} : notre système avec diacritiques [1]	92.47	94.76	97.45
UOB [ALH 07] {d}	90.96	92.95	94.44
ARAB-IFN [PEC 06] {d}	89.1	91.7	95.9
SCHMMs [BEN 06] {d}	89.79	92.25	96.78
Microsoft Research [ADB 06] {d}	88.94		95.01

TAB. 4 – Performances du reconnaisseur sans diacritiques, et comparaisons avec les principaux sytèmes de l'état de l'art.
[1] : Les performances du système après combinaison avec les signes diacritiques sont données à titre indicatif. Le détail de cette combinaison n'est pas donné dans cet article.

reur de 25% (le taux de reconnaissance passe de 89.98% à 92.47%).

Références

[ADB 06] ADBULKADER A., Two-Tier Approach for Arabic Offline Handwriting Recognition, *The Tenth International Workshop on Frontiers in Handwriting Recognition (IWFHR 10)*, La Baule, France, October 2006.

[ALH 07] AL-HAJJ R., MOKBEL C., , LIKFORMAN-SULEM L., Combination of HMM-Based Classifiers for the Recognition of Arabic Handwritten Words, *icdar*, vol. 2, 2007, pp. 959-963.

[BEN 06] BENOUARETH A., ENNAJI A., , SELLAMI M., Semi-Continuous HMMs with Explicit State Duration Applied to Arabic Handwritten Word Recognition, *The Tenth International Workshop on Frontiers in Handwriting Recognition (IWFHR 10)*, La Baule, France, 2006.

[CHE 07] CHERIET M., Strategies for visual Arabic Handwriting Recognition : issues and case study, *ISSPA 2007, International Symposium on Signal Processing and its Applications, 12 - 15 February 2007, Sharjah, United Arab Emirates*, Feb 2007.

[DUP 03] DUPRE X., Contributions à la reconnaissance de l'écriture cursive à l'aide de modèles de Markov cachés, PhD thesis, Univ Rene Descartes - Paris V, 2003.

[HIL 69] HILDITCH J., Linear skeletons from square cupboards, *Machine Intelligence 4 (B. Meltzer and D. Michie, Eds.)*, , 1969, pp. 404-420.

[KNE 98] KNERR S., AUGUSTIN E., A neural network-hidden Markov model hybrid for cursive word recognition, *Pattern Recognition, 1998. Proceedings. Fourteenth International Conference on*, vol. 2, 16-20 Aug 1998, pp. 1518-1520 vol.2.

[LOR 06] LORIGO L. M., GOVINDARAJU V., Offline Arabic Handwriting Recognition : A Survey, *IEEE Trans. Pattern Anal. Mach. Intell.*, , 2006.

[MÄR 05] MÄRGNER V., PECHWITZ M., , ABED H. E., Arabic Handwriting Recognition Competition, *ICDAR*, 2005, pp. 70–74.

[MÄR 06] MÄRGNER V., ABED H. E., , PECHWITZ M., Offline Handwritten Arabic Word Recognition Using HMM - a Character Based Approach without Explicit Segmentation, *CIFED*, 2006.

[MÄR 07] MÄRGNER V., ABED H. E., Arabic Handwriting Recognition Competition, *ICDAR*, 2007, pp. 1274-1278.

[MEN 07] MENASRI F., VINCENT N., CHERIET M., , AUGUSTIN E., Shape-Based Alphabet for Off-line Arabic Handwriting Recognition, *ICDAR '07 : Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR 2007) Vol 2*, 2007, pp. 969–973.

[PEC 02] PECHWITZ M., MADDOURI S. S., MÄERGNER V., ELLOUZE N., , AMIRI H., IFN/ENIT-DATABASE OF HANDWRITTEN ARABIC WORDS, *CIFED*, 2002.

[PEC 06] PECHWITZ M., MÄERGNER W., , ELABED H., Comparison of Two Different Feature Sets for Offline Recognition of Handwritten Arabic Words, *IWFHR*, 2006.